MD5 ALGORITHM

- المقدمة :

اليوم سوف أشرح خاورزم مشهور جدا حاليا و هو الأكثر استخداما في كثير من الأمور ، حيث يستخدم في تشفير كلمات المرور في أنظمة التشغيل و قواعد البيانات و هناك أيضا شكل مهم جدا من أشكال استخدامه و هو أساسا سبب عمل و ابتكار هذا الخواريزم ألا و هو المصداقية أي مصداقية البيانات و اسمها العلمي Data integrity و اللذي عند طريق نتيجه التشفير نعرف هل البيانات قد وصلت صحيحة أم خاطئة والخطأ يتمثل في تغيير في البينات بالزيادة أو النقصان أو التبديل على حد سواء ، و نلاحظ فائدة عندما نقوم بتحميل الملفات عن طريق برامج التورينت حيث كثير منا يحتاج **ضمان** اكتمال الملف المراد تنزيله ولا يحتمل هذا الملف أي أخطاء مثلا أننا نقوم بتنزيل نظام تشغيل فأي خطأ في الملف قد يتسبب في عدم عمل نظام التشغيل بشكل صحيح أو عطب الملف كاملا .

- ✓ طول مخرجات التشفير بالـ MD5 هو 128-bit و طول الرسالة الأصلية هو 512-bit و إن زادت فإنه يقوم بتقسيمها إلى Blocks و سنعرفك كيف يكون شكل ذلك بالرسم و التوضيح ولا يوجد حد أقصي لحجم الرسالة الأصلية أي أي أنك لو أدخلت رسالة حجمها 10GiB سيقبلها الMD5 و سيقسمها إلى Blocks.
- ✓ تحذير من خطأ شائع : كثير ما نسمع كلمة هاش أو Hash و نستخدمها في تداول البيانات لضمان صحت البيانات و نفرق دوما بين الهاش و الـ MD5 و هذا خطأ فادح فالـ MD5 هو خوارزم من خوارزمات الـ Hash لأن أول ما طرأت فكرة مصداقية البيانات قامو بعمل الـ Hash حيث استخدم الهاش بعدها في **Message Authentication Codes (MAC)** و أيضا في ال **Digital Signature** ولن نتوسع في هذا فالهاش طريق ليس له نهاية . إذن فمن اليوم عرفنا أن الـ MD5 هو شكل من أشكال خوارزمات الـ Hash .
- ✓ إني **أضمن** لك إن فهمت هذا الخوازم فهم جيدا فستفهم التالي بدون أن يشرح لك أحدا :MD4 , SHA-1 , RIPEMD-160 , HMAC .
- ✓ عندما قامو بتصميم معادلة الهاش أطلقوا عليها One Way Function أي أنهم أرادوا أن يقولوا لن نحتاج و لن نستطيع فك هذه الشفرة لنعيدها إلى شكلها الطبيعي ... طبعا أنت تقول و ما الفائدة من تشفير ملف و عدم استطاعتنا من استرجاع؟ و سأجاوب عليك بأن أقول : لماذا نسيت أنني قلت في البداية أن أساس تصميم الهاش هو لضمان **مصداقية البيانات** وليس تشفيرها ؟

و بكلامي سأضرب لك مثلا:

تخيل أنني قمت بإرسال رسالة إليك مع أحد من الناس مكتوب فيها

السلام عليكم ورحمة الله

ميعادنا اليوم إن شاء الله الساعة ١٠.٣٠ **مساء**

تحياتي .

تخيل أن الذي ينقل رسالتي إليك قام بتعديل و تزوير الرسالة و أصبحت هكذا :
السلام عليكم ورحمة الله
ميعادنا اليوم إن شاء الله الساعة ١٠.٣٠ **صباحا**
تحياتي .
أرأيت ؟ ... الموضوع خطير جدا

ماذا لو أرسلت الرساله بهذه الطريقة التالية :
١/ أرسلت إليك معه رسالة محتواها التالي :
MD5 = d3eb64402c360513d2842ce53cf20e41
٢/ ثم بعدها أرسلت معه هذه الرسالة طبعا هو لم يعلم أني سأكتب هذه اكلام في هذه المرة و قد نقل الرسالة الأولى بدون معرفة الرسالة الثناية :
السلام عليكم ورحمة الله
ميعادنا اليوم إن شاء الله الساعة ١٠.٣٠ مساء
تحياتي .

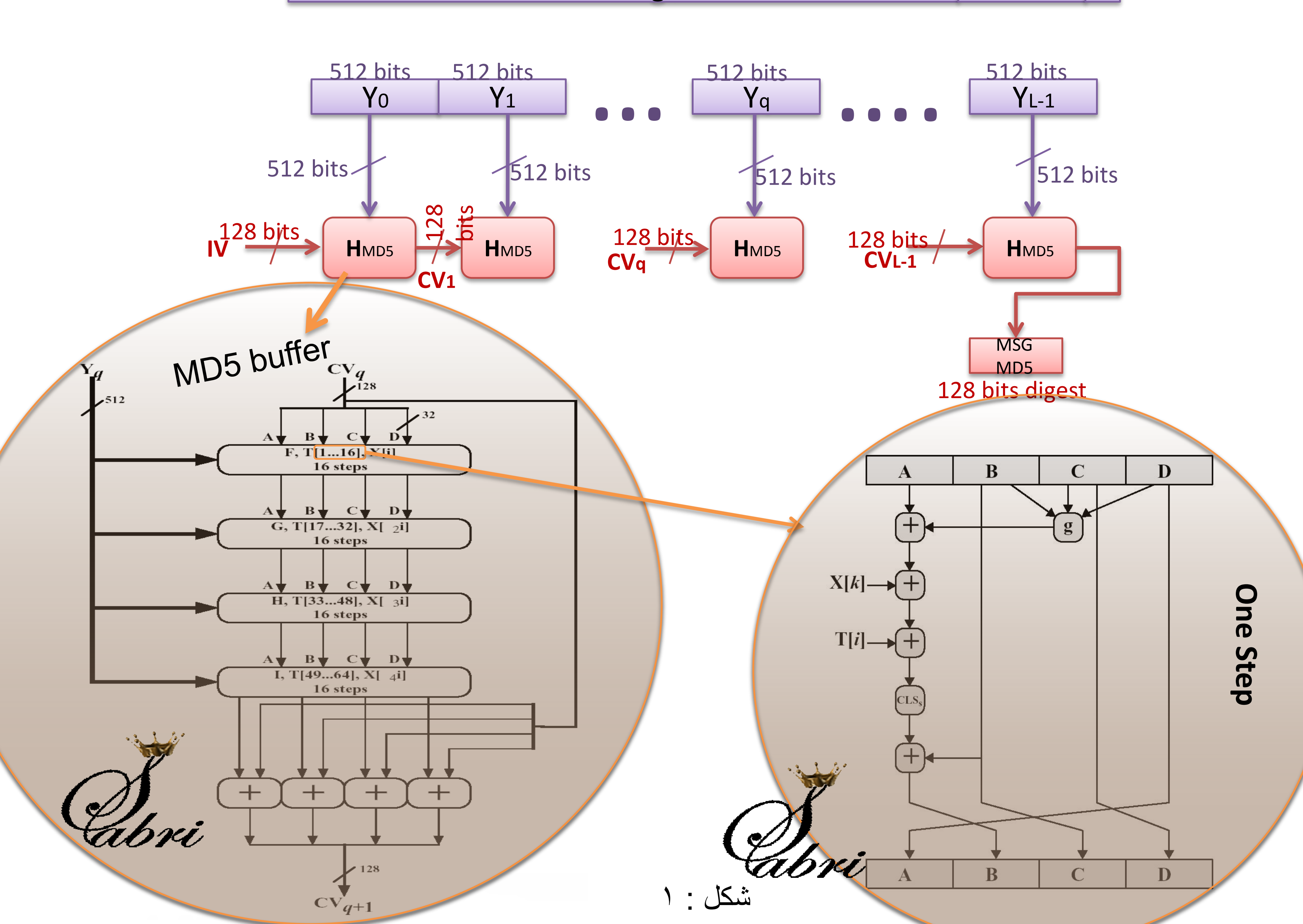
الأن لا يستطيع أن يغير في الرسالة ولو غيير التغيير اللذي فرضناه في المثال السابق
السلام عليكم ورحمة الله
ميعادنا اليوم إن شاء الله الساعة ١٠.٣٠ **صباحا**
تحياتي .

ثم قمت أنت بمقارنة الـ MD5 اللذي أرسلته لك سابقا
ستجد أن ال MD5 للرسالة المزورة هو 5aedd5a4d1ab6bbe5df190c0b767f73e

هناك فرق ؟ نعم هنا فرق و أظنك إستوعبت الفكرة جيدا .

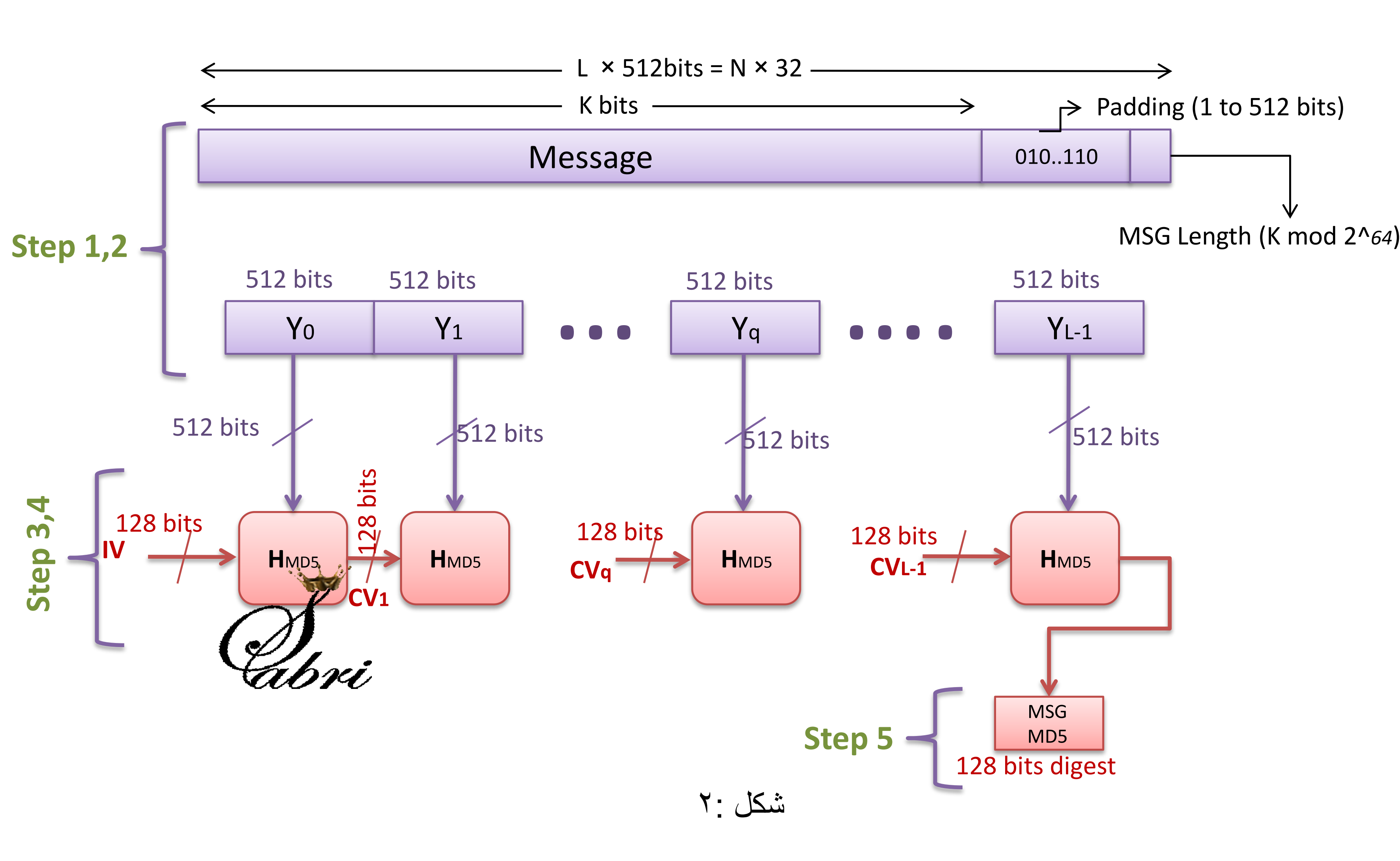
و كما تعودنا سأقوم برسم الرسومات كاملة ثم أقوم بتفصيلها .

لنبدأ بسم الله ..



Message
010..110
512 bits
Y0
512 bits
Y1
512 bits
Yq
512 bits
YL-1
512 bits
512 bits
512 bits
512 bits
IV
128 bits
HMD5
128 bits
CV1
HMD5
128 bits
CVq
HMD5
128 bits
CVL-1
HMD5
MSG
MD5
128 bits digest
MD5 buffer
Yq
512
CVq
128
32
A B C D
F, T[1…16], X[i]
16 steps
G, T[17…32], X[2i]
16 steps
H, T[33…48], X[3i]
16 steps
I, T[49…64], X[4i]
16 steps
128
CVq+1
A B C D
g
X[k]
T[i]
CLSs
One Step
Sabri

شكل : ١



شكل :٢

❑ شكل خارجي للرسالة عند معالجتها مع معادلة MD5 لكي تخرج لنا القيمة المتعلقة بالرسالة بتشفير الـ MD5

إن عملية الـ Hashing تمر بخمس خطوات جميلة و سهلة الفهم لمن لدية الخلفية في مبادئ التشفير فإني أرى أنها لا تزيد صعوبتها على من تتوافر لديهم هذا الطلب اللذي قد اشترطه في شرح الـ DES آنفا. سأسرد الخطوات بدون تفصيل أولا ثم سأفسر عند الحاجة لها في الرسمة طبعا سنشرح الخطوات كلها إن شاء الله و لكنني أحب أن أجعل خارطتك الذهني واضحة و مترابطة منذ البداية (لنرى الرسومات مع بعضها و ترابطها مع بعضها و لنرى كل النقاط مسرودة ثم يأتي التفصيل) فهذي طريقتي في المذاكرة وهذه طريقتي في الشرح .

- ✓ Step 1 : Append Padding bits
- ✓ Step 2 : Append Length
- ✓ Step 3 : Initialize MD buffer
- ✓ Step 4 : Process Message in 512-bit (16-word) blocks
- ✓ Step 5 : Output

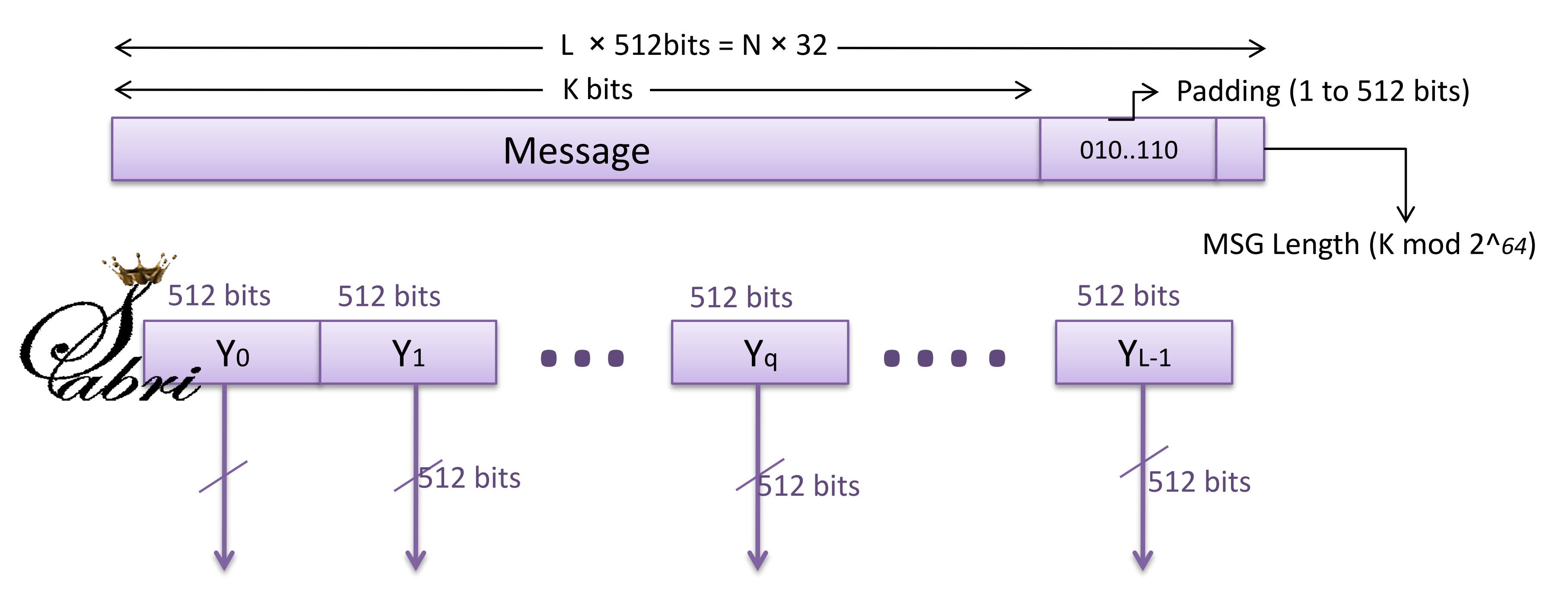
الأن لإكمال جذور الشجرة الذهنية سأضع الرسمة و ما تقدمه و تعمله من خطواتنا الخمس بالتفصيل .

ملاحظات لضمان للإستفادة المتكاملة :

- لم أضع الصور بدقة كبيرة لتنظر إليها و هي مصغرة في الموضوع فالتصغير لترتيب الموضوع و الدقة الكبيرة لترى كل التفاصيل أي أنني أطلب منك أخي الحبيب أن تعرض الصور مكبرة قدر المستطاع .
- عند إحساسك بغياب الصورة الشاملة الأولى(الشكل:١و ٢) عن ذهنك في وسط الشرح الرجاء الرجوع و إلقاء نظرة أخرى عليها و ستلاحظ في كلامي أنني قد أعيد بعض المعلومات لكي يتم الربط بين المعلومات بشكل شجري صحيح .
- عندما أطلب منك الرجوع لرسمة معينة فأرجو أن تنظر إلى **كل الرسمة** اللتي حدتها لك <u>بتأني</u> ولا تبحث بعينيك على الحرف أو الرمز المتعلق بالموضوع فقط فكل رسوماتنا متعلقة جدا ببعضها .
- إن الترميز في الكتابة يؤثر بشكل ملحوظ على النواتج و أقصد بالترميز هو مثلا .. Asci و UTF-8 و ANSI و UCS حيث كل ترميز أو Encoding يحول الحروف إلى HEX بقيم مختلفة عن الأخر مما يغير شكل المدخلات في عملياتنا و أظنك فهمت ما أقصد .

هيا لنفهم معنى الشكل :٢ و اللتي ضم كل الخطوات لكي نفهم التفاصيل بعد ذلك خطوة خطوة مع الرسم .

إن خوارزم الـ MD5 كأي خوارزم تشفير له قالب ذو مساحه معينة يقوم بتخزين الرسالة فيه و أقصد طبعا بالرسالة هي المدخلات اللتي نريد أن نخرج لها Hash بعد عملية المعالجة و أيضا نفس الكلام ينطبق على المخرجات فهي تخضع لحجم محدد لا يزيد و لا نقص ، فبالنسبة للـ MD5 من ناحة المدخلات فهو يتحمل طول رسالة تصل 512-bit فإن زادت عن ذلك فإنه يقوم بتقسيم الرسالة إلى أكثر من قالب أو بمعنى علمي أكثر من Block و إن نقصت عن 512-bit فإنه يخضع للمعالجة بعمليات الـ Padding و الجدير بالذكر أن المخرجات ستكون 128-bit فقط طالت أم قصرت الرسالة ولذلك فهو تشفير طول مفاتحه 128-bit و لا يجب أن ننسى ذلك ، وسنبدأ نتكلم عن العمليات بالتفصيل .



شكل : ٣

✓Step1: Append Padding bits:-

قد ذكرنا في الـ DES كلمة Padding و قلنا أنها عبارة مجموعة من الـ Bits توضع في أخر الرسالة إذا كان طول الرسالة أقصر من الـ Message Block size اللذي يتحمله هذا النوع من التشفير أيا كان نوعه و لا بد من وجوده على الأقل و لو 1-bit و على الأكثر 512-bit طبعا هذه الأرقام خاصة بالـ MD5 لكن ال Padding توجود في كل أنواع التشفير باختلافي مساحة الـ Padding Block و قلنا أننا لن نبحر في تعريف الـ Padding لكننا سنتكلم عنه هنا فقط فيما يخص MD5 .

لننظر إلى الـ Frame الأول في الرسمة(الشكل:٣) سنجد أنه يوجد ثلاثة أقسام :

Message : و هو المكان اللذي تخزن فيه رسالتنا أو مدخلاتنا .

Padding : و هو جزءنا اللذي نشرحه و نرى أن أقصى طول له 512-bit اي أنه لا يوجد رسالة في الأصل و بالتالي فإن الجزء الأخير Message Lengthلا يحمل أي قيم .

MSG Length : أي Message Length و هو جزء يتم تحديد طول الرسالة الأصلية قبل وضع الـ Padding .

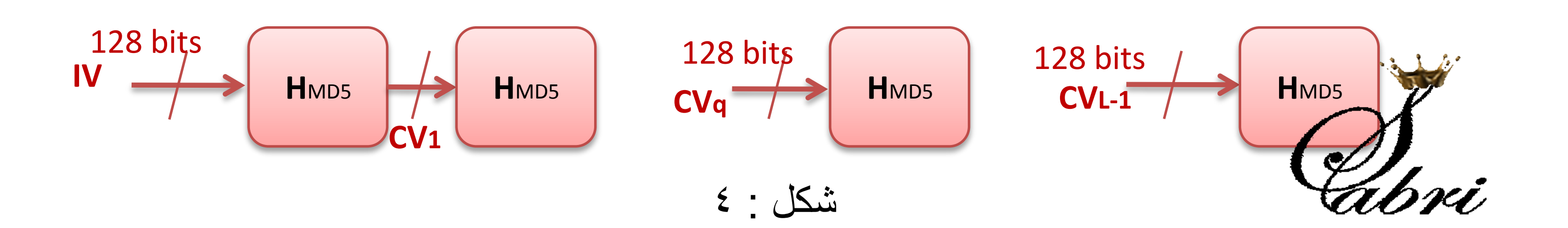
إذن لو فرضنا أن طول رسالتنا الحقيقي هو 448-bits فإن طول الـ Padding سيساوي 512 – 448 و النتيجه 960-bits طبعا القيمة 448 تشمل الــMSG Length لأننا ذكرنا أنا توضع قبل حساب الـ Padding .

✓Step 2: Append Length:-

هي عملية حساب طول الرسالة الحقيقي و يتم حجز فيه 64-bit مباشرة بمجرد وضع الرسالة و قبل وضلع الـ Padding و الواجب ذكره أنه هذه 64-bit ليست آخر قيمة نستطيح وضعها لطول الراسلة الأصلية كاملة ففي الحقية نستطيع وضع حتى 2^{64} bit كقيمة لطول الرسالة الحقيقية و طبعا هذه قيمة كبيرة جدا تدل على أن الرسالة كبيرة جدا و يجب تقسيمها على أكثر من Block كما هو موضح في الرسمة في المربعات اللتي تحت المربع الكبير الأول فكل Block حجمة لا يزيد عن 512-bit (Y_0, Y_1 , ... Y_q , ... Y_{L-1}) ، وبحكم أننا في جزء نتحدث فيه عن طول الرسالة فقط آن أوان توضيح بعض الرموز اللتي على الرسمة ...

512bits × L هو طول الرسالة الحقيقية كاملة و L هو عدد الـBlocks اللتي تم تقسيم الرسالة عليها و تتضمن أيضاً الPadding و Message Length .

✓ Step 4: Initialize MD buffer:-



شكل : ٤

في المربعات الحمراء (الشكل:٤) يظهر لنا الـ MD5 buffer و اللذي يستقبل أو يخزن في أول buffer من اليسار أربع قيم محفوظين في Registers حجم كل واحد منها 32-bit أي أن مجموعهم 128-bit و هو ما اتفقنا عليه آنفا أن طول مفتاح ال MD5 هو 128-bit تسمى أول قيمة بـ Initial Value أو IV و قد سميت هذه الـ Registers بـ A , B , C , D على التوالي و تحمل القيم التالية بالـ Hexadecimal

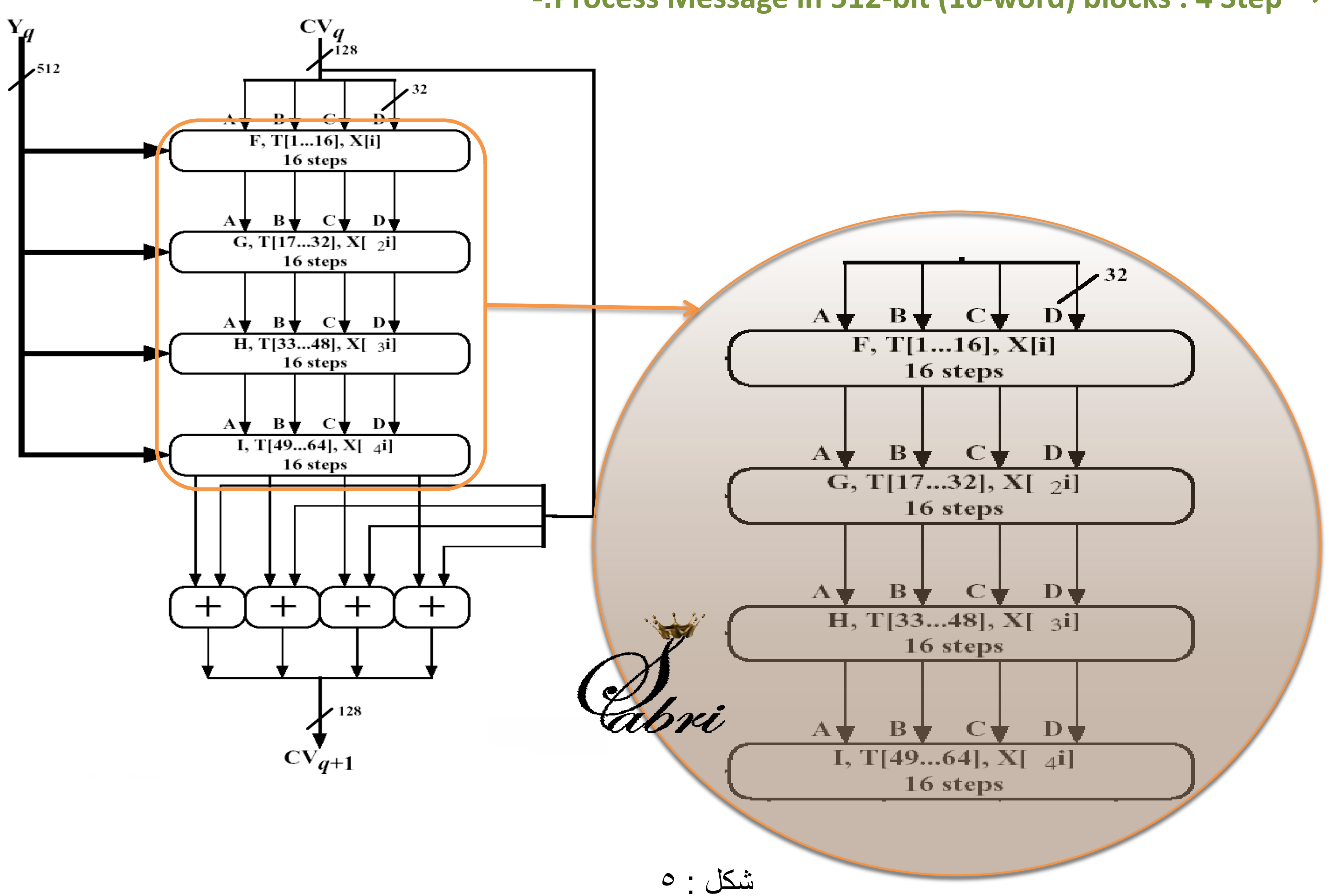
A = 67452301

B = EFCDAB89

C = 98BADCFE

D = 10325476

✓ -:Process Message in 512-bit (16-word) blocks : 4 Step



شكل : ٥

إن قلب أي خوارزم هو تلك المربعات اللتي تضاف إلي الرسالة تقوم بعملية التشفير فمثلا عندما تريد أن تقوم بتقطيع ورقة عادية فإن قلب عملية التقطيع هو المقص أو الأداة اللتي ستقوم بتقطيع الورقة فوجب علينا التبحر في أداتنا اللتي ستضاف إلى رسالتنا لتخرج لنا التشفير و يجب أن نعرف ما هي مكوناتها كلها لكي نكون قد عرفنا فعلا ما اللذي يحدث في MD5 ليخرج لنا هذه النتائج و هذه الخطوة هو قلب موضوعنا و أتمنى منك أخي القارء التريث و التركيز في هذه الجزئية.

من الرسمة السابقة(الشكل:٥) نجد هناك أربع مستطيلات فوق بعضها و تسمى الـ Round في كل Round تدخل البيانات في معالجة تمر بـ 16 خطوة أي 16 Steps كل الخطوات لها نفس الخوارزم و سنوضح خواريزم الخطوة الواحدة و اللذي هو نفس الخوارزم في كل الخطوات و الفائدة من هذه الخطوات طبعا هو زيادة تعقيد التشفير ، لنبدأ بشرح الـ Roundأولا ثم بعدها نتغلغل داخلها لنرى الـ Steps .

✓ الـ Rounds :

F, T[1...16], X[i]
16 steps

- هناك أربع Rounds في كل مربع مكتوب عليه H_{MD5} موجود في الرسمة (شكل:١) فكلهم لهم نفس البناء الخوارزمي كما ذكرنا ولكن تختلف في أنها تستخدم عدد أولي منطقي مختلف مثل F,G,H,I فقيم تلك الأعداد مختلفة (إنتبه: سنعود لنعلق على هذه الجملة الأخيرة قريبا جدا فاجعلها في هامش ذاكرتك الشجرية لكي نربطها بشكل مرتب) و بنائها الداخلي قلنا هو عبارة عن ١٦ Step
- كل Round تأخذ مدخلات من حجمها 512-bit من Y_q (الشكل:٥) حيث **قيمة** Y_q هي الرسالة المقسمة و قد عرفنا أن الرسالة تقسم إلى Blocks كل واحد حجمة 512-bit (الشكل:٣)
- تدخل قيمة الـ Y_q واللتي حجمها 512-bit على Round مع دخول أيضا قيم الـ Append Length أي A,B,C,D و اللتي مجموع قيمهم 128-bit ... هل تذكرهم ؟ راجع شجرتك الذهنية (الشكل:٤) و يجب أن نعلم أن من كلامنا أن قيمة Y_q تكون ثابتة في كل الأربع Rounds لـ H_{md5} buffer الواحد و تختلف طبعا من كل H_{md5} buffer و اللذي يليه بسبب إختلاف قيمة جزء الرسالة .
- نرى في ال Rounds T[1 ...16] و T[17 ...32] و T[33 ...48] و T[49 ...64] أي أن T تأخذ أرقام متغيرة و سنسندها إلى متغر نرمز له بالرمز i حيث i قيمته عدد صحيح من ١ إلى ٦٤ و كل رقم له قيمة ثابتة <u>**بالراديان**</u> في جدول رمزنا له بالرمز T و الأن القارئ يقول إذن ماهي T[i] ؟ فأجيبك من كلامي السابق هي القيمة بالراديان المسندة للرقم i في الجدول T . لكن ... مازال من حقتك أن تعرف في ماذا و كيف عرفت أنا هذه القيم اللتي في الجدول هل هي من رأسك ؟ أم تخضع لمعادلة أم ماذا ؟

فأجيبك و أقول لا ليس من رأسي بل تخضع لمعادلة و هي:

$$2^{32} \times |\sin(i)|$$

حيث : sin(i) قيمته ما بين واحد و صفر و كما قلنا القيمة بالراديان و | | تحني أن القيمة مطلقة أي أننا نتجاهل إشارة السالب إن وجدت

وللتسهيل .. فإننك بفضل اله لن تحتاج لحساب كل قيم ال T[i] و سأضح لك جدولا فيه كل قيم T[i] لكي ننهي هذه المسألة .. إليك الجدول :

T[1] = D76AA478	T[17] = F61E2562	T[33] = FFFA3942	T[49] = F4292244
T[2] = E8C7B756	T[18] = C040B340	T[34] = 8771F681	T[50] = 432AFF97
T[3] = 242070DB	T[19] = 265E5A51	T[35] = 699D6122	T[51] = AB9423A7
T[4] = C1BDCEEE	T[20] = E9B6C7AA	T[36] = FDE5380C	T[52] = FC93A039
T[5] = F57C0FAF	T[21] = D62F105D	T[37] = A4BEEA44	T[53] = 655B59C3
T[6] = 4787C62A	T[22] = 02441453	T[38] = 4BDECFA9	T[54] = 8F0CCC92
T[7] = A8304613	T[23] = D8A1E681	T[39] = F6BB4B60	T[55] = FFEFF47D
T[8] = FD469501	T[24] = E7D3FBC8	T[40] = BEBFBC70	T[56] = 85845DD1
T[9] = 698098D8	T[25] = 21E1CDE6	T[41] = 289B7EC6	T[57] = 6FA87E4F
T[10] = 8B44F7AF	T[26] = C33707D6	T[42] = EAA127FA	T[58] = FE2CE6E0
T[11] = FFFF5BB1	T[27] = F4D50D87	T[43] = D4EF3085	T[59] = A3014314
T[12] = 895CD7BE	T[28] = 455A14ED	T[44] = 04881D05	T[60] = 4E0811A1
T[13] = 6B901122	T[29] = A9E3E905	T[45] = D9D4D039	T[61] = F7537E82
T[14] = FD987193	T[30] = FCEFA3F8	T[46] = E6DB99E5	T[62] = BD3AF235
T[15] = A679438E	T[31] = 676F02D9	T[47] = 1FA27CF8	T[63] = 2AD7D2BB
T[16] = 49B40821	T[32] = 8D2A4C8A	T[48] = C4AC5665	T[64] = EB86D391

✓ الـ Step :

الأن جاء دور التعمق في الـ Round و قلنا أن الـ MD5 فيه ٤ Rounds و قد شرحنا الشكل الخارجي للـ Round فيا ترى ، ما هو الشكل الدخالي لها ؟ أجيبك الشكل الداخلي هو عبادة عن الـ ١٦ Step اللتي قد نوهنا عنها بدون تفصيل و قلنا أنها تحمل أن كل الـ Steps لها نفس الخوارزم و لكن فقط تختلف في قيم الأعداد الأولية **F,G,H,I** و اللتي قد قلنا سنعود لها لاحقا (انظر النقطة الأولى في تعريف الـ Round) ، إن كل Round تمتلك ترتيبا تسلسليا من الـ Steps موجودة الـ Buffer قلنا أننا أطلقنا عليها الرموز التالية A B C D و قد وضحنا أنها تحمل قيم أيضا (انظر Step3: Initialize MD buffer)

إن تلك الـ Steps تخضع لمعادلة سهلت علينا الكثير في حساب قيمتها .. إليك المعادلة

$$a \leftarrow b + ((a + g(a, b, c) + X[k] + T[i]) <<< s)$$

حيث :

a, b, c, d = الـ buffer اللذي فيهل قيم الأربع الأولية بترتيب محدد في كل Step

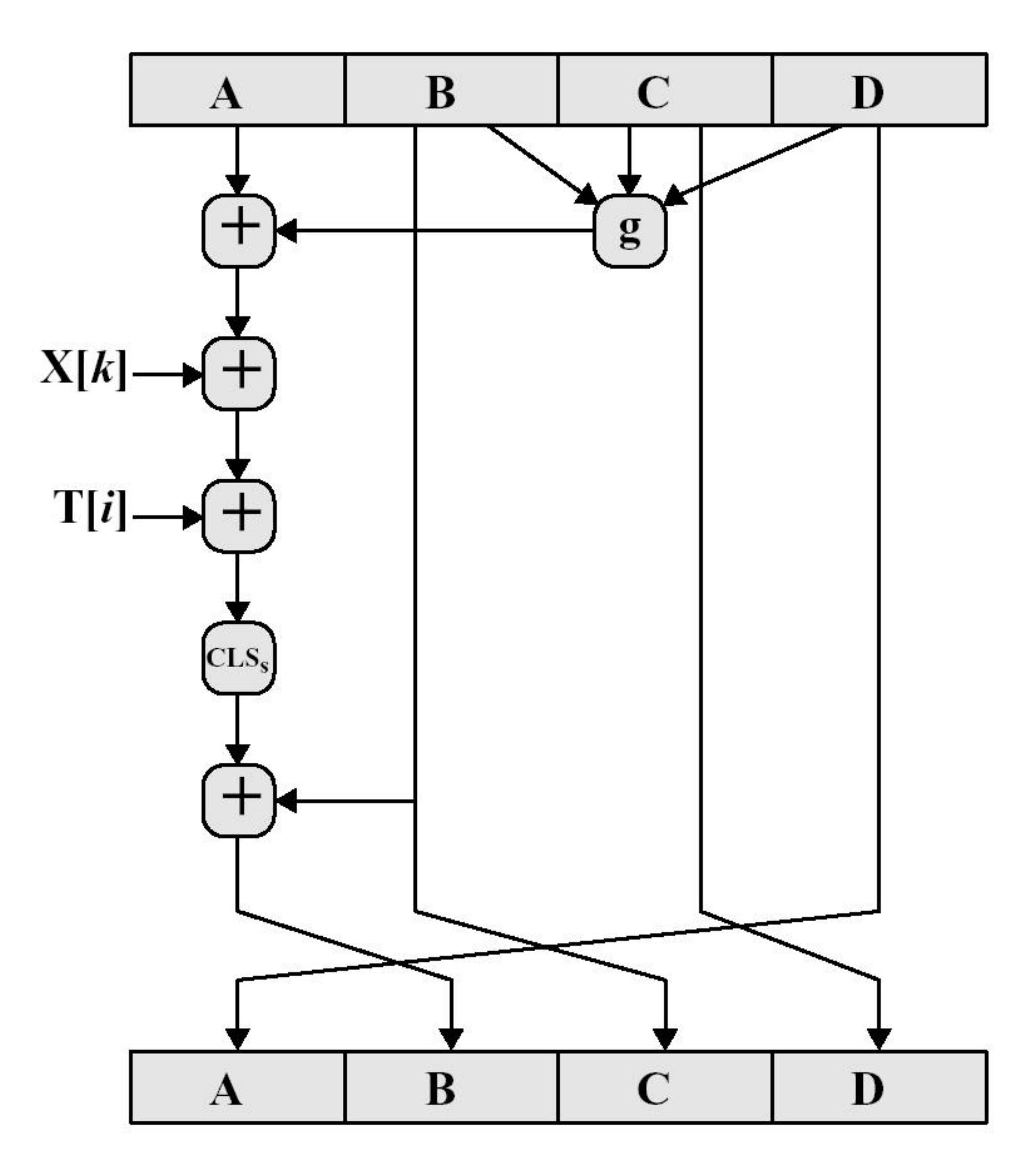
g = واحدة من القيمة الأولية الأخرى F, G , H , I أي نها متغير .

s <<< = مقدار إزاحة الـ Bits من إزاحة من اليسار إلى اليمين حيث s هو مقدار الإزاحة و <<< هو اتجاه الإزاحة، مثال : 00110 لو أزحناها بمقدار ٢ من اليسار إلى اليمين ستصبح النتيجة 11000

X[k] = M[q × 16 +k] =

T[i] = و قد عرفنا شرحها سابقا (انظر النقطة الرابعة في تعريف الـ Round)

وخوارزم هذه المعادلة السابقة نجده في الشكل : ٦ انظر إليه ثم عد وانظر في المعادلة :



شكل : ٦

- واحدة من المتغيرات الأولية g و اللذي هي كما قلنا عبارة عن واحده من F,G,H,I تستخدم في كل Round من الـ ٤ Rounds الموجده في خوارزم MD5 .
- كل متغير أولي يأخذ ثلاث مدخلات أو ثلاثة words وهم من الـ Registers b,c,d حجم كل مدخل 32-bit و نفس الحجم في المخرجات و لن نستخدم ال Register A في المدخلات لكننا سنستخدمها لاحقا و سترون ذلك (انظر الشكل:٦) .
- إن شكل حساب المدخلات مع المتغيرات الأولية يخضع لمعادلة منطقية خاصة بكل Round على حدى .
- دعنا نلخص المعادلات مع أماكنها في كل Round :

Round	Primitive Function g	g(b, c, d)
1	F(b, c, d)	(b ∧ c) v (!b ∧ d)
2	G(b, c, d)	(b ∧ d) v (c ∧ !d)
3	H(b, c, d)	b ⊕ c ⊕ d
4	I(b, c, d)	c ⊕ (b v !d)

سأوضح معنى العمليات :

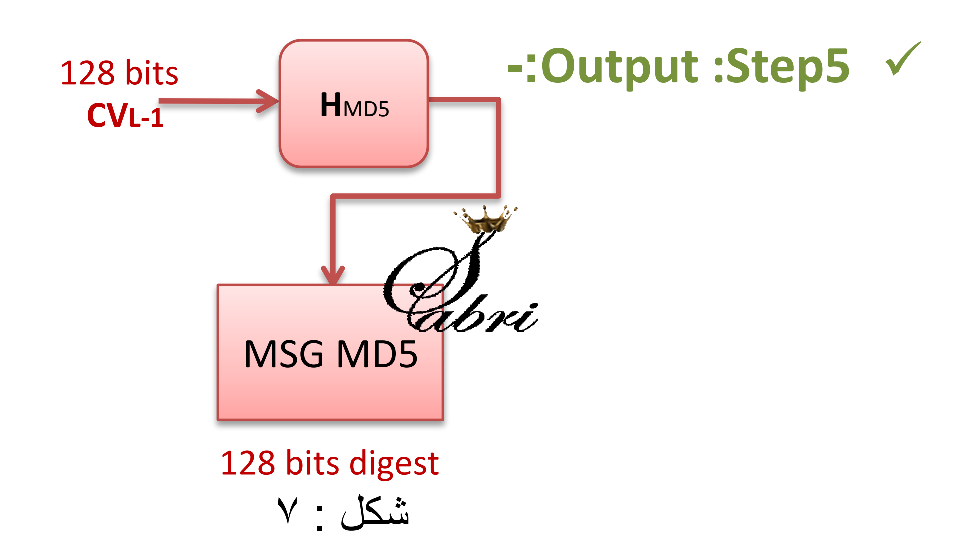
AND = Λ

OR = V

NOT = !

XOR = ⊕

✓ Step5: Output:-



شكل : ٧

بعد إتمام كل العمليات السابقة على كل أجزاء الرسالة **L** فإن المخرجات ستكون عبارة عن 128-bit و هنا ستمى هذا الـ128-bit بالــ **Message Digest**

نستطيع تلخيص كل عمليات الـ MD5 في هذه المعادلة الرائعة و اللتي أفهمتني الكثير من الموضوع بشكل شخصي

$CV_0 = IV$

$CV_{q+1} = SUM_{32} [CV_q , RF_I (Y_q , RF_H , (Y_q , RF_G (Y_q , RF_F (Y_q , CV_q))))]$

$MD = CV_{L-1}$

حيث :

IV = قيمة ابتدائية تدخل على ABCD buffer في الخطوة رقم ٣

Y_q = رقم ترتيب الرسالة المجزأه إلى 512-bit

L = رقم ترتيب رسالة ولكن متضمنة (processed with the qth block of message(padding and Length

CV = **Channing Variable** و هي ال Blocks Y_q اللتي تم عليها المعالجة من الرسالة.

RF_x = **Round Function** و هذه المعادلة تستخدم المعادلات الحسابية المنطقية .

SUM 23 = **Addition modulo 2^{32}** وهي عملية جمع منطقية من نوع خاص تستطيع حسابها بالألة العلمية المتقدمة في الكمبيوتر

MD = **Message Digest** و هي قيمة أو ناتج الرسالة بعد التشفير .

بعد كل هذا الإنهاك في هذا الموضوع القوي و اللذي إن فهمته ستفهم الكثير الكثير كما ذكرت في المقدمة

يتبادر إلى ذهنك سؤال .. كيف نستطيع كسر هذا التشفير المعقد و القوي جدا ؟ و الجواب .. الطريقة هي بطريقى مقارنة الهاش بجداول ضخمة من الكلمات اللذي يقابلها الـMD5 الخاص بها ... لم تفهم ؟ دعنا نقول مثال عملي سهل جدا

إذا كنت قد وضعت كلمة سر مثلا و هي

Sabri فإن الـ Md5 لها ناتجه هو a40ee2f2e5f22604be74c2a2b5fd11d5

الأن أنت تريد أن تخمن فستخمن أولا أني قد وضعت إسمي ككلمة مرور ولكنك لا تعرف كيف كتبت الكلمة فسجرب مثلا

Sbri و ناتجها ستجده : e6d337f75d38c4b016647aa1afe183f2

Sabary ناتجها ستجده : dd8ebd456b7a6f28e1c900f16bbe8b45

Sabry ناتجها ستجده : 47f5816e677805626ffca1d9a7e782fc

Sabre ناتجها ستجده : dca095c8198150f390922fb9fbada318

Sapry ناتجها ستجده : 9edb424c54488d7682b8c0158cea8d80

Sapri ناتجها ستجده : afb8413192b42f2b79e854b3bb8f97dd

لاحظت ؟ .. غير مطابقة فإن وجدت كلمة Sabri بهذا الشكل فسيقارن النواتج و ستجدها متطابقة و عندها تكون نجحت في معرفة أو كسر التشفير فالموضوع تخمين الكلمة و مقارنة الـ MD5

لتطبيق درسنا الدسم تطبيقا عمليا علميا

وفقني الله تعالي في أن أجد هذه الصفحة في موقع تعليمي يختص بتعليم السكيورتي بطريقة عليمة عملية رائعة

تفضل طبق على درسنا ..
http://nsfsecurity.pr.erau.edu/crypto/md5.html

البرامج

على بيئة الويندوز : برنامج **Fsum**

طريقة عمله من الـ DOS

fsum.exe <OPTIONS> <FILE>

```
C:\WINDOWS\system32\cmd.exe
C:\>fsum.exe -md5 KING.txt

SlavaSoft Optimizing Checksum Utility - fsum 2.52.00337
Implemented using SlavaSoft QuickHash Library <www.slavasoft.com>
Copyright (C) SlavaSoft Inc. 1999-2007. All rights reserved.

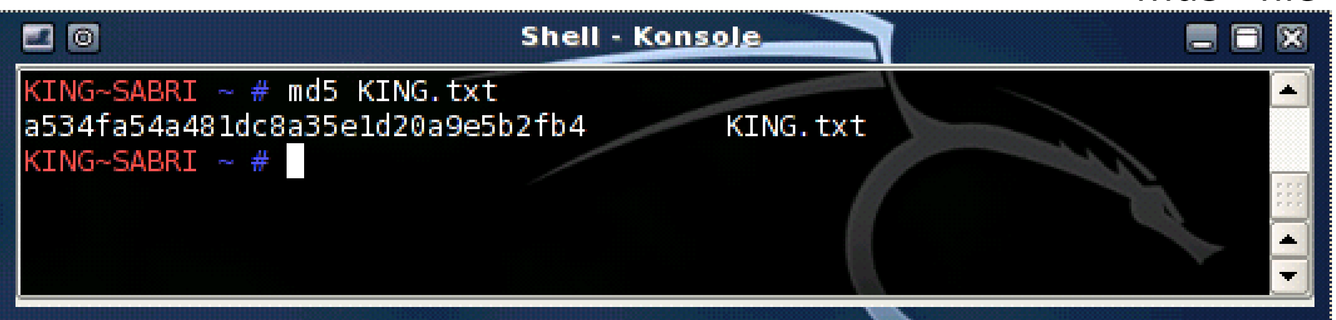
; SlavaSoft Optimizing Checksum Utility - fsum 2.52.00337 <www.slavasoft.com>
;
; Generated on 12/18/08 at 05:04:40
;
d5a1ca5fae44e4c849a8a900cdcc015f *KING.txt

C:\>
```

على بيئة لينوكس : برنامج **MD5**

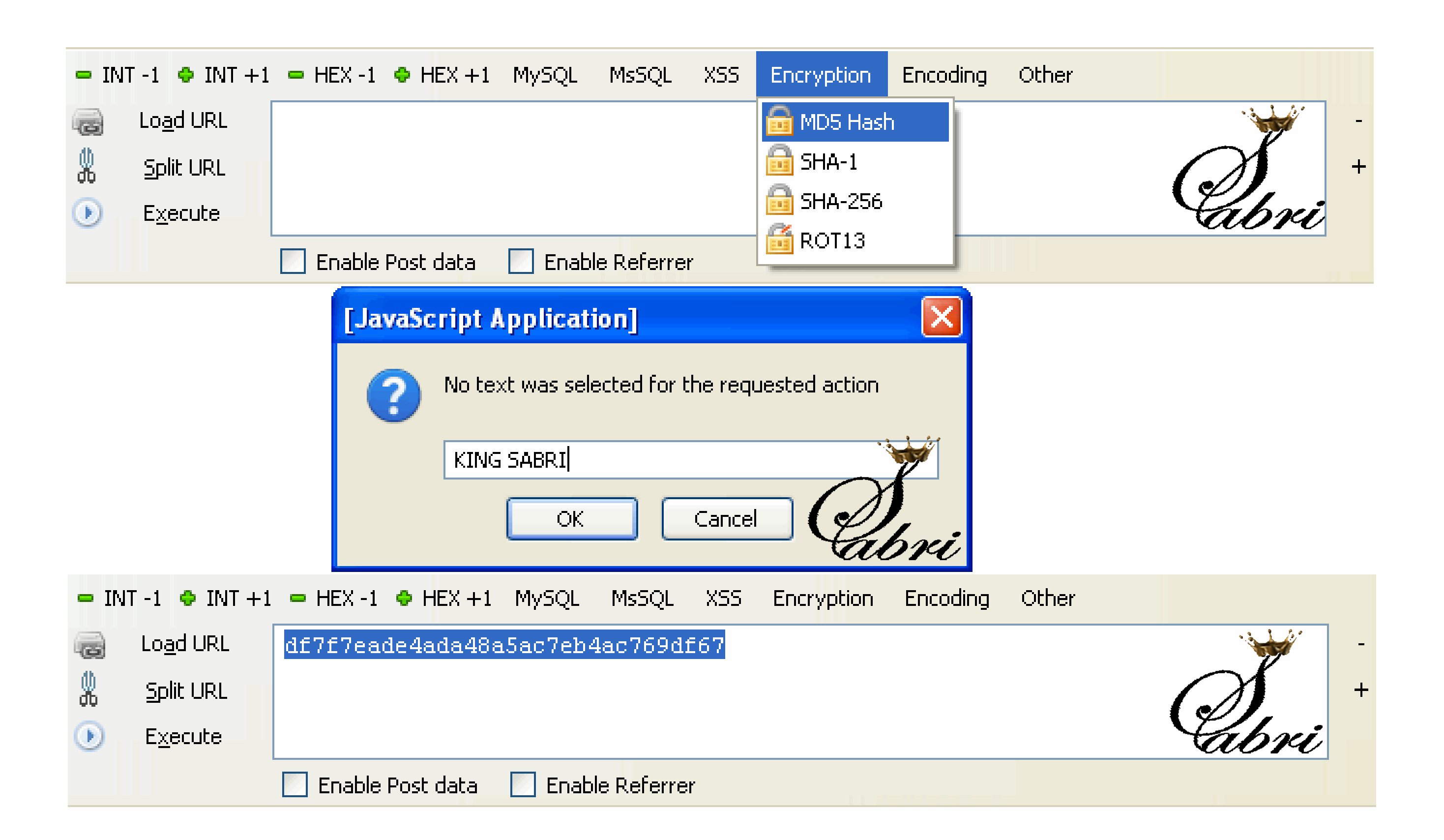
طريقه عمله من الـ Terminal

md5 <file>



طبعا لا نكتب " <> " عند التطبيق

على بيئة المتصفح Firefox (على الويندوز و اللينوكس) : إضافة **HackBar**



المراجع :

- حبيب قلبي William Stallings و كتابه الأشهر بين نظرائه Cryptography and Network Security **الإصدار الثالث** .
- http//king-sabri.net

للمراسلة عبر الإيميل وليس عبر الماسنجير : king-sabrii@hotmail.com

أسألكم الدعاء لي ولوالدي ...

